

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-176187

(43)Date of publication of application : 13.07.1993

(51)Int.Cl.

H04N 1/419

G06F 15/66

H03M 7/30

(21)Application number : 03-338874

(71)Applicant : FUJITSU LTD

(22)Date of filing : 20.12.1991

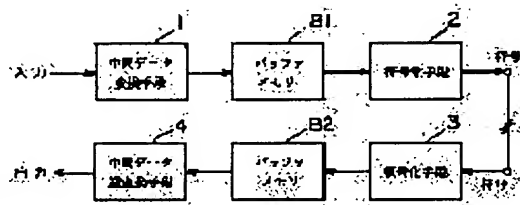
(72)Inventor : OKADA YOSHIYUKI  
YOSHIDA SHIGERU  
NAKANO YASUHIKO  
CHIBA HIROTAKA

## (54) DATA COMPRESSION DECODING DEVICE

## (57)Abstract:

PURPOSE: To improve the compression and decoding processing speed for the entire device.

CONSTITUTION: The data compression decoding device is made up of a data compression circuit consisting of an intermediate data conversion means 1, a coding means 2, and a buffer memory B1, and up of a data decoding circuit consisting of a decoding means 3, an intermediate data inverse conversion means 4 and a buffer memory B2. The intermediate data conversion means 1 in the data compression circuit converts input data into intermediate data and stores the data tentatively into the buffer memory B1. The coding means 2 codes the intermediate data stored in the buffer memory B1 into an output code. The decoding means 3 in the data decoding circuit converts the input code into intermediate data and stores the data tentatively to the buffer memory B2. The intermediate data inverse conversion means 4 inverts the intermediate data stored in the buffer memory B2 into output data.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 ( J P )

(12) 公開特許公報 ( A )

(11) 特許出願公開番号

特開平 5 - 1 7 6 1 8 7

(43) 公開日 平成 5 年 ( 1 9 9 3 ) 7 月 1 3 日

(51) Int. Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
H04N 1/419		8839-5C		
G06F 15/66	330	E 8420-5L		
H03M 7/30		8836-5J		

審査請求 未請求 請求項の数 7 (全 2 9 頁)

(21) 出願番号 特願平 3 - 3 3 8 8 7 4

(22) 出願日 平成 3 年 ( 1 9 9 1 ) 1 2 月 2 0 日

(71) 出願人 0 0 0 0 0 5 2 2 3

富士通株式会社

神奈川県川崎市中原区上小田中 1 0 1 5 番  
地

(72) 発明者 岡田 佳之

神奈川県川崎市中原区上小田中 1 0 1 5 番  
地 富士通株式会社内

(72) 発明者 吉田 茂

神奈川県川崎市中原区上小田中 1 0 1 5 番  
地 富士通株式会社内

(74) 代理人 弁理士 井桁 貞一

最終頁に続く

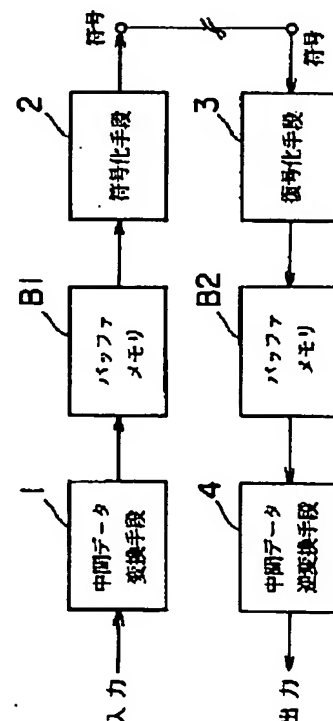
(54) 【発明の名称】 データ圧縮・復元装置

(57) 【要約】

【目的】 データ圧縮・復元装置に関し、装置全体の圧縮及び復元処理速度を向上させることを目的とする。

【構成】 データ圧縮・復元装置は、中間データ変換手段 1、符号化手段 2 及びバッファメモリ B 1 から構成するデータ圧縮回路、及び復号化手段 3、中間データ逆変換手段 4 及びバッファメモリ B 2 から構成するデータ復元回路からなる。データ圧縮回路では、中間データ変換手段 1 が入力データを中間データに変換し、バッファメモリ B 1 に一時的に格納する。符号化手段 2 がバッファメモリ B 1 に格納された中間データを出力符号に符号化する。データ復元回路では、復号化手段 3 が入力符号を中間データに変換し、バッファメモリ B 2 に一時的に格納する。中間データ逆変換手段 4 がバッファメモリ B 2 に格納された中間データを出力データに逆変換する。

本発明のデータ圧縮・復元装置の原理説明図



1

## 【特許請求の範囲】

【請求項 1】 情報源から入力された入力データを、符号化することにより圧縮するデータ圧縮回路において、入力データを中間データに変換する中間データ変換手段（１）と、

前記中間データを出力符号に符号化する符号化手段（２）と、

前記中間データ変換手段（１）と前記符号化手段（２）とに、直列に接続されるバッファメモリ（Ｂ１）と、を有することを特徴とするデータ圧縮回路。

【請求項 2】 情報源から入力された入力データを、符号化することにより圧縮するデータ圧縮回路において、入力データを中間データに変換する中間データ変換手段（１）と、

前記中間データを出力符号に符号化する符号化手段（２）と、

前記入力データを一時的に格納するバッファメモリ（Ｂ１）と、前記中間データを一時的に格納するバッファメモリ（Ｂ１）と、前記出力符号を一時的に格納するバッファメモリ（Ｂ１）とを、前記入力データの性質に応じて設ける少なくとも二以上のバッファメモリ（Ｂ１）と、

を有することを特徴とするデータ圧縮回路。

【請求項 3】 情報源から入力された入力データを、符号化することにより圧縮するデータ圧縮回路において、入力データを中間データに変換する複数の中間データ変換手段（１）と、

前記中間データ変換手段（１）に対応して設けられ、前記中間データを一時的に格納するバッファメモリ（Ｂ１）と、

前記複数の中間データ変換手段（１）から出力された前記中間データを選択する選択手段（１１）と、

前記選択手段（１１）によって選択された前記中間データを出力符号に符号化する符号化手段（２）と、

前記選択手段（１１）によって選択された前記中間データを積算して前記選択手段（１１）に出力するランレングス積算手段（１２）と、

を有することを特徴とするデータ圧縮回路。

【請求項 4】 情報源から入力された入力符号を、復号化することにより復元するデータ復元回路において、入力符号を中間データに変換する復号化手段（３）と、前記中間データを出力データに逆変換する中間データ逆変換手段（４）と、

前記復号化手段（３）と前記中間データ逆変換手段（４）とに、直列に接続されるバッファメモリ（Ｂ２）と、

を有することを特徴とするデータ復元回路。

【請求項 5】 情報源から入力された入力符号を、復号化することにより復元するデータ復元回路において、入力符号を中間データに変換する復号化手段（３）と、

2

前記中間データを出力データに逆変換する中間データ逆変換手段（４）と、

前記入力符号を一時的に格納するバッファメモリ（Ｂ２）と、前記中間データを一時的に格納するバッファメモリ（Ｂ２）と、前記出力データを一時的に格納するバッファメモリ（Ｂ２）とを、前記入力符号の性質に応じて設ける少なくとも二以上のバッファメモリ（Ｂ２）と、

を有することを特徴とするデータ復元回路。

10 【請求項 6】 情報源から入力された入力符号を、復号化することにより復元するデータ復元回路において、入力符号を中間データに変換する復号化手段（３）と、前記中間データを出力データに逆変換する複数の中間データ逆変換手段（４）と、

前記複数の中間データ逆変換手段（４）に対応して設けられ、前記中間データを一時的に格納するバッファメモリ（Ｂ２）と、

前記複数の中間データ逆変換手段（４）から出力された前記出力データを選択する選択手段（１３）と、

20 前記復号化手段（３）によって出力された前記中間データを積算して前記選択手段（１３）に出力するランレングス積算手段（１４）と、

を有することを特徴とするデータ復元回路。

【請求項 7】 情報源から入力された入力データ及び入力符号を、符号化及び復号化することによりデータの圧縮及び復元をするデータ圧縮・復元装置において、入力データを中間データに変換する中間データ変換手段（１）と、前記中間データを出力符号に符号化する符号化手段（２）と、前記中間データ変換手段（１）と前記

30 符号化手段（２）とに、直列に接続されるバッファメモリ（Ｂ１）とからなるデータ圧縮回路と、

入力符号を中間データに変換する復号化手段（３）と、前記中間データを出力データに逆変換する中間データ逆変換手段（４）と、前記復号化手段（３）と前記中間データ逆変換手段（４）とに、直列に接続されるバッファメモリ（Ｂ２）とからなるデータ復元回路と、

を有することを特徴とするデータ圧縮・復元装置。

## 【発明の詳細な説明】

【 0 0 0 1 】

40 【産業上の利用分野】 本発明はデータ圧縮・復元装置に関し、特にランレングス変換法を使用してデータの圧縮及び復元を行うデータ圧縮・復元装置に関する。

【 0 0 0 2 】 近年、著しい技術開発によって、様々のＯＡ機器は飛躍的な発展を遂げつつある。しかし、ファクシミリや光ディスクファイル・システム等のように、ＯＡ機器で文書情報を取り扱う場合、白黒二値画像のデータ量は文字情報に比べて非常に大きく、十数倍から数十倍にもなる。また、ＯＡ機器の一つであるファクシミリでは、画像の品位を向上させるために従来のＧ３機（解像度 200dpi）からＧ４機（解像度 400dpi）へ移行しつつ

50

ある。したがって、従来のデータ量の増加以上にデータ量が増加しつつある。このようなデータ量の大幅な増加に対処するため、データの内容を損なわずに全体のデータ量を減らす方式、すなわちデータ圧縮方式が提案されている。

【0003】なお、「文字 (Character)」及び「文字列 (Character String)」の定義は J I S - C 6 2 3 0 に従うほか、情報理論で用いられている呼称を踏襲し、1ワード単位で構成されるデータを「文字」と呼び、任意のワード単位で構成されるデータを「文字列」と呼ぶことにする。

【0004】

【従来の技術】従来、白黒二値画像の圧縮方式としては、MMR (Modified Modified READ) 方式及び予測符号化方式が知られている。この一例として、本出願人による特開平3-58574号公報があり、データの性質に依存せず、簡単な方式で効率よく圧縮するデータ圧縮方式を開示している。このデータ圧縮方式は、二次元的白黒二値画像情報を前処理によって取り入れ、この前処理によって二次元的白黒二値画像情報をランレングス変換し、さらにユニバーサル符号化方式によって符号化するものである。

【0005】ユニバーサル符号化方式は、データ圧縮時に情報源の統計的な性質を予め仮定しないため、文字コード、オブジェクトコード等の種々のデータタイプに適用することができる。そして、ユニバーサル符号化方式の代表的な例として、L Z (Lempel-Ziv) 符号化法と算術符号化法とがある。また、L Z 符号化法には、ユニバーサル型と増分分解型 (Incremental persing) のアルゴリズムが提案されている。さらに、これらのアルゴリズムを改良した符号化法として、ユニバーサル型に属する L Z S S 符号化法と、増分分解型に属する L Z W (Lempel-Ziv-Welch) 符号化法とがある。

【0006】なお、L Z 符号化法は、例えば、宗像清治著「Lempel-Zivデータ圧縮法」、情報処理、pp.2~6, Vol.26, No.1, 1985 に詳しく掲載されている。また、L Z S S 符号化法は、T.C. Bell, "Better OPM/L Text Compression", IEEE Trans. on Commu., Vol.COM-34, No.12, Dec.1986 に詳しく掲載されている。さらに、L Z W 符号化法は、T.A. Welch, "A Technique for High-Performance Data Compression", Computer, Jun.1984 に詳しく掲載されている。そして、増分分解型の符号化法及び L Z W 符号化法は、特開昭59-231683号、米国特許No. 4,558,302号に開示されている。

【0007】これらの符号化法のうち、高速処理ができ、アルゴリズムが簡単であるという利点から、一般的に L Z W 符号化法が使用されてきた。L Z W 符号化法は、書き換え可能な辞書をもち、以下に示す処理によって符号化を行う方法である。まず、新規の入力文字列を相異なる部分文字列に分割し、この部分文字列が辞書に

登録されてなければ、出現した順に識別番号を付して全て辞書に登録する。同時に、現在入力している部分文字列のうち、最長の部分文字列と一致する部分文字列を辞書から選択し、選択した部分文字列に付されている識別番号で符号化する。

【0008】以下、L Z W 符号化法を使用したデータ圧縮回路及びデータ復元回路について説明する。図27は、従来の処理回路の概略ブロック図である。図27

(A) は従来のデータ圧縮回路の概略ブロック図を、図27 (B) は従来のデータ復元回路の概略ブロック図を示す。

【0009】図27 (A) において、他の装置からデータ圧縮回路への入力速度を L、中間データ変換手段 121 の入力速度を  $M_i$ 、中間データ変換手段 121 の出力速度を  $M_o$ 、ユニバーサル符号化手段 122 の入力速度を  $N_i$ 、ユニバーサル符号化手段 122 の出力速度を  $N_o$ 、及びデータ圧縮回路から他の装置への出力速度を P とする。ここで、中間データ変換手段 121 の入力速度  $M_i$ 、出力速度  $M_o$ 、ユニバーサル符号化手段 122 の入力速度  $N_i$ 、出力速度  $N_o$  は、いずれも符号化処理するデータの性質によって処理速度が異なる。なお、速度の単位はいずれも [bytes/sec] である。

【0010】この場合、データ圧縮回路への入力速度 L と中間データ変換手段 121 の入力速度  $M_i$  との関係において、

入力速度 L > 入力速度  $M_i$

ならば、圧縮回路へデータを入力する他の装置が待機させられる。同様に、中間データ変換手段 121 の出力速度  $M_o$  とユニバーサル符号化手段 122 の入力速度  $N_i$  との関係において、

出力速度  $M_o$  > 入力速度  $N_i$

ならば、中間データ変換手段 121 の出力側が待機させられる。特に、中間データ変換手段 121 の入力速度  $M_i$ 、出力速度  $M_o$  について、

平均:  $M_o = M_i / 2$ , 最小:  $M_o = M_i / 8$ , 最大:  $M_o = 2 M_i$

なる関係があるため、

出力速度  $M_o$  < 入力速度  $N_i$

の場合には、ユニバーサル符号化手段 122 の入力側が待機させられるため、全体の処理速度も低下する。

【0011】同様に、図27 (B) において、他の装置からデータ復元回路への入力速度を Q、ユニバーサル復号化手段 123 の入力速度を  $R_i$ 、ユニバーサル復号化手段 123 の出力速度を  $R_o$ 、中間データ逆変換手段 124 の入力速度を  $S_i$ 、中間データ逆変換手段 124 の出力速度を  $S_o$ 、及びデータ復元回路から他の装置への出力速度を T とする。ここで、中間データ逆変換手段 124 の入力速度  $S_i$ 、出力速度  $S_o$ 、ユニバーサル復号化手段 123 の入力速度  $R_i$ 、出力速度  $R_o$  は、いずれも復号化処理するデータの性質によって処理速度が異なる。

る。なお、速度の単位はいずれも〔bytes/sec〕である。

【0012】この場合、データ復元回路への入力速度 $Q$ とユニバーサル復号化手段123の入力速度 $R_i$ との関係において、

入力速度 $Q > \text{入力速度 } S_i$

ならば、復元回路へデータを入力する他の装置が待機させられる。同様に、ユニバーサル復号化手段123の出力速度 $R_o$ と中間データ逆変換手段124の入力速度 $S_i$ との関係において、

出力速度 $R_o > \text{入力速度 } S_i$

ならば、中間データ逆変換手段124の出力側が待機させられる。特に、中間データ逆変換手段124の入出力速度 $S_i$ 、 $S_o$ について、

平均： $S_o = 2 S_i$ 、 最小： $S_o = 8 S_i$ 、 最大： $S_o = S_i / 2$

なる関係があるため、

出力速度 $R_o > \text{入力速度 } S_i$

の場合には、ユニバーサル復号化手段123の出力側が待機させられるため、全体の処理速度も低下する。

【0013】

【発明が解決しようとする課題】しかし、従来のデータ圧縮回路及びデータ復元回路では、データ圧縮処理及びデータ復元処理を行うと、中間データ変換手段及びユニバーサル符号化手段の処理速度はデータの性質によって大きく影響される。このため、各手段の前後において処理速度が異なり、他の手段は処理データが到着しないために待機しなければならない。

【0014】このように、いずれかの手段が待機させられると、その待機時間が累積し、中間データに変換する処理とユニバーサル符号化する処理とが並行して行うことができなくなった。このため、データ圧縮回路及びデータ復元回路の回路全体の処理速度が低下するという問題点があった。

【0015】本発明はこのような点に鑑みてなされたものであり、データ圧縮回路及びデータ復元回路全体の処理速度を向上させるデータ圧縮・復元装置を提供することを目的とする。

【0016】

【課題を解決するための手段】本発明では上記目的を達成するために、図1に示すように、データ圧縮回路は中間データ変換手段1、符号化手段2及びバッファメモリB1から構成する。中間データ変換手段1は入力された入力データを中間データに変換する。バッファメモリB1は中間データ変換手段1及び符号化手段2と直列に接続され、中間データ変換手段1から出力された中間データを一時的に格納する。符号化手段2はバッファメモリB1に格納された中間データを出力符号に符号化する。

【0017】また、図5に示すように、データ圧縮回路は中間データ変換手段1、符号化手段2及び少なくとも

二以上のバッファメモリB1から構成する。中間データ変換手段1は入力された入力データを中間データに変換する。バッファメモリB1は中間データ変換手段1及び符号化手段2と直列に接続され、入力データ、中間データ変換手段1から出力された中間データ、あるいは出力符号のいずれか少なくとも二以上のデータを一時的に格納する。符号化手段2は中間データを出力符号に符号化する。

【0018】さらに、図7に示すように、データ圧縮回路は複数の中間データ変換手段1、複数のバッファメモリB1、符号化手段2、選択手段11及びランレングス積算手段12から構成する。中間データ変換手段1は入力された入力データを中間データに変換する。バッファメモリB1は中間データ変換手段1に対応して設けられ、中間データを一時的に格納する。符号化手段2は中間データを出力符号に符号化する。選択手段11は、ランレングス積算手段12からの積算結果に基づき、中間データ変換手段1から出力された中間データのいずれかを選択する。ランレングス積算手段12は、選択手段11によって選択された中間データを入力して積算し、その積算結果を選択手段11に出力する。

【0019】そして、図9に示すように、データ復元回路は復号化手段3、中間データ逆変換手段4及びバッファメモリB2から構成する。復号化手段3は、入力された入力符号を中間データに変換する。バッファメモリB2は復号化手段3及び中間データ逆変換手段4と直列に接続され、復号化手段3から出力された中間データを一時的に格納する。中間データ逆変換手段4は、バッファメモリB1に格納された中間データを出力データに逆変換する。

【0020】それから、図13に示すように、データ復元回路は復号化手段3、中間データ逆変換手段4及び少なくとも二以上のバッファメモリB2から構成する。復号化手段3は入力符号を中間データに復号化する。バッファメモリB2は復号化手段3及び中間データ逆変換手段4と直列に接続され、入力符号、復号化手段3から出力された中間データ、あるいは出力データのいずれか少なくとも二以上のデータを一時的に格納する。中間データ逆変換手段4は中間データを出力データに逆変換する。

【0021】そのうえ、図15に示すように、データ復元回路は復号化手段3、複数の中間データ逆変換手段4、複数のバッファメモリB2、選択手段13及びランレングス積算手段14から構成する。復号化手段3は入力符号を中間データに復号化する。バッファメモリB2は中間データ逆変換手段4に対応して設けられ、復号化手段3から出力された中間データを一時的に格納する。中間データ逆変換手段4は中間データを出力データに逆変換する。選択手段13は、ランレングス積算手段14からの積算結果に基づき、中間データ逆変換手段4から

出力された出力データのいずれかを選択する。ランレングス積算手段 1 4 は、復号化手段 3 から出力された中間データを入力して積算し、その積算結果を選択手段 1 3 に出力する。

【 0 0 2 2 】そして、図 1 7 に示すように、データ圧縮・復元装置は、データ圧縮回路及びデータ復元回路から構成する。データ圧縮回路は中間データ変換手段 1、符号化手段 2 及びバッファメモリ B 1 から構成する。中間データ変換手段 1 は入力された入力データを中間データに変換する。バッファメモリ B 1 は中間データ変換手段 1 及び符号化手段 2 と直列に接続され、中間データ変換手段 1 から出力された中間データを一時的に格納する。符号化手段 2 はバッファメモリ B 1 に格納された中間データを出力符号に符号化する。また、データ復元回路は復号化手段 3、中間データ逆変換手段 4 及びバッファメモリ B 2 から構成する。復号化手段 3 は、入力された入力符号を中間データに変換する。バッファメモリ B 2 は復号化手段 3 及び中間データ逆変換手段 4 と直列に接続され、復号化手段 3 から出力された中間データを一時的に格納する。中間データ逆変換手段 4 は、バッファメモリ B 1 に格納された中間データを出力データに逆変換する。

【 0 0 2 3 】

【作用】データ圧縮回路において、中間データ変換手段 1 は入力された入力データを中間データに変換し、バッファメモリ B 1 に一時的に格納する。符号化手段 2 は、バッファメモリ B 1 に格納された中間データを出力符号に符号化する。

【 0 0 2 4 】また、データ圧縮回路において、入力された入力データをバッファメモリ B 1 に一時的に格納する。中間データ変換手段 1 は入力データを中間データに変換し、バッファメモリ B 1 に一時的に格納する。符号化手段 2 は、バッファメモリ B 1 に格納された中間データを出力符号に符号化し、バッファメモリ B 1 に一時的に格納する。

【 0 0 2 5 】さらに、データ圧縮回路において、入力された入力データを所定量ごとに複数の中間データ変換手段 1 が中間データに変換し、対応するバッファメモリ B 1 に一時的に格納する。選択手段 1 1 がランレングス積算手段 1 2 の積算結果に基づき、複数のバッファメモリ B 1 に格納された中間データを選択し、符号化手段 2 が選択された中間データを出力符号に符号化する。

【 0 0 2 6 】そして、データ復元回路において、復号化手段 3 は入力された入力符号を中間データに変換し、バッファメモリ B 2 に一時的に格納する。中間データ逆変換手段 4 は、バッファメモリ B 2 に格納された中間データを出力データに逆変換する。

【 0 0 2 7 】それから、データ復元回路において、入力された入力符号をバッファメモリ B 2 に一時的に格納する。復号化手段 3 は入力符号を中間データに変換し、バ

ッファメモリ B 2 に一時的に格納する。中間データ逆変換手段 4 は、バッファメモリ B 2 に格納された中間データを出力データに逆変換し、バッファメモリ B 2 に一時的に格納する。

【 0 0 2 8 】そのうえ、データ復元回路において、復号化手段 3 は入力された入力符号を中間データに変換する。この中間データは、選択手段 1 3 がランレングス積算手段 1 4 の積算結果に基づき、選択されたバッファメモリ B 2 に一時的に格納される。中間データ逆変換手段 4 は対応するバッファメモリ B 2 に格納された中間データを出力データに逆変換して出力する。

【 0 0 2 9 】そして、データ圧縮・復元装置において、データ圧縮回路では、中間データ変換手段 1 は入力された入力データを中間データに変換し、バッファメモリ B 1 に一時的に格納する。符号化手段 2 は、バッファメモリ B 1 に格納された中間データを出力符号に符号化する。データ復元回路では、復号化手段 3 は入力された入力符号を中間データに変換し、バッファメモリ B 2 に一時的に格納する。中間データ逆変換手段 4 は、バッファメモリ B 2 に格納された中間データを出力データに逆変換する。

【 0 0 3 0 】

【実施例】以下、本発明の一実施例を図面に基づいて説明する。図 2 は、本発明のデータ圧縮回路の第 1 の実施例を示す図である。図において、データ圧縮回路はバッファメモリ B 1 a、ランレングス変換手段 1 a 及び L Z W 符号化手段 2 a から構成される。

【 0 0 3 1 】バッファメモリ B 1 a は、他の装置から入力された入力データを一時的に格納する。ランレングス変換手段 1 a は、入力データを中間データであるランレングスに変換する。L Z W 符号化手段 2 a は、ランレングスを出力符号に符号化する。なお、ランレングス変換手段 1 a は図 1 の中間データ変換手段 1 に、L Z W 符号化手段 2 a は図 1 の符号化手段 2 に、それぞれ相当する。

【 0 0 3 2 】したがって、バッファメモリ B 1 a を設けたことにより、他の装置及びランレングス変換手段 1 a は、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ圧縮回路全体の処理速度が向上する。

【 0 0 3 3 】図 3 は、本発明のデータ圧縮回路の第 2 の実施例を示す図である。図において、データ圧縮回路はランレングス変換手段 1 a、バッファメモリ B 1 b 及び L Z W 符号化手段 2 a から構成される。ここで、図 2 と同一の要素には同一番号を付し、説明を省略する。

【 0 0 3 4 】バッファメモリ B 1 b は、ランレングス変換手段 1 a によって出力されたランレングスを一時的に格納する。したがって、バッファメモリ B 1 b を設けたことにより、ランレングス変換手段 1 a 及び L Z W 符号化手段 2 a は、互いに他の手段の処理速度に影響される

10

20

30

40

50

ことがなく各々の手段の処理を行うことができる。このため、データ圧縮回路全体の処理速度が向上する。

【0035】図4は、本発明のデータ圧縮回路の第3の実施例を示す図である。図において、データ圧縮回路はランレングス変換手段1a、バッファメモリB1c及びLZW符号化手段2aから構成される。ここで、図2と同一の要素には同一番号を付し、説明を省略する。

【0036】バッファメモリB1cは、LZW符号化手段2aによって出力された出力符号を一時的に格納する。したがって、バッファメモリB1cを設けたことにより、LZW符号化手段2a及び他の装置は、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ圧縮回路全体の処理速度が向上する。

【0037】図6は、本発明のデータ圧縮回路の第4の実施例を示す図である。図において、データ圧縮回路はランレングス変換手段1a、バッファメモリB1a、B1b、B1c及びLZW符号化手段2aから構成される。ここで、図2、図3及び図4と同一の要素には同一番号を付し、説明を省略する。

【0038】したがって、バッファメモリB1a、B1b、B1cを設けたことにより、いずれの手段及び他の装置は、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ圧縮回路全体の処理速度が向上する。

【0039】なお、バッファメモリB1a、B1b、B1cは、入力される入力データの性質及びコストに応じて、いずれのバッファメモリを設置したり、設置しなかったりすることもできる。これにより、入力データに最適なデータ圧縮回路を構成することができる。

【0040】図8は、本発明のデータ圧縮回路の第5の実施例を示す図である。図において、データ圧縮回路はランレングス変換手段1a、1b、バッファメモリB1a、B1b、選択手段11、ランレングス積算手段12及びLZW符号化手段2aから構成される。

【0041】ランレングス変換手段1a、1bは、入力された入力データを中間データであるランレングスに変換する。バッファメモリB1a、B1bは、それぞれランレングス変換手段1a、1bによって出力されたランレングスを一時的に格納する。選択手段11は、ランレングス積算手段12の演算結果に基づき、バッファメモリB1a又はバッファメモリB1bのいずれかに格納されたデータを入力するために切り換える。ランレングス積算手段12は、選択手段11によって選択されたバッファメモリの処理データ数を積算し、この積算結果を選択手段11に出力する。LZW符号化手段2aは、選択手段11によって選択されたバッファメモリに格納されたランレングスを出力符号に符号化する。

【0042】したがって、ランレングス変換手段1aと選択手段11との間にバッファメモリB1aを設け、ラ

ンレングス変換手段1bと選択手段11との間にバッファメモリB1bを設けたことにより、ランレングス変換手段1a、1b及びLZW符号化手段2aは、互いに他の手段の処理速度に影響されることがなく各々の手段の処理を行うことができる。

【0043】また、ランレングス変換手段1a、1bの処理速度を大幅に超える入力データが入力された場合でも、選択手段11によっていずれかのランレングス変換手段に切り換えて変換処理することができる。このため、データ圧縮回路全体の処理速度がより向上する。

【0044】なお、ランレングス変換手段1a、1bは図1の中間データ変換手段1に、バッファメモリB1a、B1bは図1のバッファメモリB1に、LZW符号化手段2aは図1の符号化手段2に、それぞれ相当する。

【0045】また、データ圧縮回路の第5の実施例では、ランレングス変換手段及びバッファメモリは2組設けたが、入力データの速度に応じて3組以上のランレングス変換手段及びバッファメモリを設けてもよい。これにより、入力データの速度に最適なデータ圧縮回路を構成することができ、よりデータ圧縮回路全体の処理速度が向上する。

【0046】そして、LZW符号化手段2aは一つしか設けていないが、ランレングス変換手段及びバッファメモリに対応して設けてもよい。それから、ランレングス変換手段1aと選択手段11との間にバッファメモリB1aを設け、ランレングス変換手段1bと選択手段11との間にバッファメモリB1bを設けたが、他の装置からの入力とランレングス変換手段1a、1bとの間、LZW符号化手段2aと他の装置への出力の間にバッファメモリを設けてもよい。あるいは、これらのいずれの間に少なくとも一つ以上のバッファメモリを設けてもよい。

【0047】図10は、本発明のデータ復元回路の第1の実施例を示す図である。図において、データ復元回路はバッファメモリB2a、LZW復号化手段3a及びランレングス逆変換手段4aから構成される。

【0048】バッファメモリB2aは、他の装置から入力された入力符号を一時的に格納する。LZW復号化手段3aは、入力符号を中間データであるランレングスに復号化する。ランレングス逆変換手段4aは、ランレングスを出力データに逆変換する。なお、LZW復号化手段3aは図9の復号化手段3に、ランレングス逆変換手段4aは図9の中間データ逆変換手段4に、それぞれ相当する。

【0049】したがって、バッファメモリB2aを設けたことにより、他の装置及びLZW復号化手段3aは、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ復元回路全体の処理速度が向上する。

【 0 0 5 0 】 図 1 1 は、本発明のデータ復元回路の第 2 の実施例を示す図である。図において、データ復元回路は L Z W 復号化手段 3 a、バッファメモリ B 2 b 及びランレングス逆変換手段 4 a から構成される。ここで、図 1 0 と同一の要素には同一番号を付し、説明を省略する。

【 0 0 5 1 】 バッファメモリ B 2 b は、L Z W 復号化手段 3 a によって出力されたランレングスを一時的に格納する。したがって、バッファメモリ B 2 b を設けたことにより、L Z W 復号化手段 3 a 及びランレングス逆変換手段 4 a は、互いに他の手段の処理速度に影響されることがなく各々の手段の処理を行うことができる。このため、データ復元回路全体の処理速度が向上する。

【 0 0 5 2 】 図 1 2 は、本発明のデータ復元回路の第 3 の実施例を示す図である。図において、データ復元回路は L Z W 復号化手段 3 a、バッファメモリ B 2 c 及びランレングス逆変換手段 4 a から構成される。ここで、図 1 0 と同一の要素には同一番号を付し、説明を省略する。

【 0 0 5 3 】 バッファメモリ B 2 c は、ランレングス逆変換手段 4 a によって出力された出力データを一時的に格納する。したがって、バッファメモリ B 2 c を設けたことにより、ランレングス逆変換手段 4 a 及び他の装置は、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ復元回路全体の処理速度が向上する。

【 0 0 5 4 】 図 1 4 は、本発明のデータ復元回路の第 4 の実施例を示す図である。図において、データ復元回路は L Z W 復号化手段 3 a、バッファメモリ B 2 a、B 2 b、B 2 c 及びランレングス逆変換手段 4 a から構成される。ここで、図 1 0、図 1 1 及び図 1 2 と同一の要素には同一番号を付し、説明を省略する。

【 0 0 5 5 】 したがって、バッファメモリ B 2 a、B 2 b、B 2 c を設けたことにより、いずれの手段及び他の装置は、互いの処理速度に影響されることがなく各々の処理を行うことができる。このため、データ復元回路全体の処理速度が向上する。

【 0 0 5 6 】 なお、バッファメモリ B 2 a、B 2 b、B 2 c は、入力される入力符号の性質及びコストに応じて、いずれのバッファメモリを設置したり、設置しなかったりすることもできる。これにより、入力データに最適なデータ復元回路を構成することができる。

【 0 0 5 7 】 図 1 6 は、本発明のデータ復元回路の第 5 の実施例を示す図である。図において、データ復元回路は L Z W 復号化手段 3 a、バッファメモリ B 2 a、B 2 b、ランレングス逆変換手段 4 a、4 b、選択手段 1 1 及びランレングス積算手段 1 2 から構成される。

【 0 0 5 8 】 L Z W 復号化手段 3 a は、入力された入力符号を中間データであるランレングスに復号化する。バッファメモリ B 2 a、B 2 b は、それぞれ L Z W 復号化

手段 3 a によって出力されたランレングスを一時的に格納する。ランレングス逆変換手段 4 a、4 b は、それぞれバッファメモリ B 2 a、B 2 b に格納されたランレングスを出力データに逆変換する。選択手段 1 1 は、ランレングス積算手段 1 2 の積算結果に基づき、ランレングス逆変換手段 4 a 又はランレングス逆変換手段 4 b の出力データを選択するために切り換える。また、同時に選択手段 1 1 は、使用するバッファメモリ B 2 a、B 2 b をも切り換える。ランレングス積算手段 1 2 は、L Z W 復号化手段 3 a によって出力された処理データ数に応じて積算し、積算結果を選択手段 1 1 に出力する。

【 0 0 5 9 】 したがって、バッファメモリ B 2 a、B 2 b を設けたことにより、L Z W 復号化手段 3 a 及びランレングス逆変換手段 4 a、4 b は、互いに他の手段の処理速度に影響されることがなく各々の手段の処理を行うことができる。このため、データ復元回路全体の処理速度が向上する。

【 0 0 6 0 】 なお、L Z W 復号化手段 3 a は図 1 5 の復号化手段 3 に、バッファメモリ B 2 a、B 2 b は図 1 5 のバッファメモリ B 2 に、ランレングス逆変換手段 4 a、4 b は図 1 5 の中間データ逆変換手段 4 に、それぞれ相当する。

【 0 0 6 1 】 また、データ復元回路の第 5 の実施例では、ランレングス逆変換手段及びバッファメモリは 2 組設けたが、入力データの速度に応じて 3 組以上のランレングス逆変換手段及びバッファメモリを設けてもよい。これにより、入力符号の速度に最適なデータ復元回路を構成することができ、よりデータ復元回路全体の処理速度が向上する。

【 0 0 6 2 】 そして、L Z W 復号化手段 3 a は一つしか設けていないが、ランレングス逆変換手段及びバッファメモリに対応して設けてもよい。それから、ランレングス逆変換手段 4 a、4 b の処理速度を大幅に超える入力データが L Z W 復号化手段 3 a から出力された場合でも、選択手段 1 1 によっていずれかのランレングス逆変換手段に切り換えて変換処理することができる。L Z W 復号化手段 3 a とランレングス逆変換手段 4 a、4 b との間にバッファメモリ B 2 a、B 2 b を設けたが、他の装置からの入力と L Z W 復号化手段 3 a との間、選択手段 1 1 と他の装置への出力の間にバッファメモリを設けてもよい。あるいは、これらのいずれの間に少なくとも一つ以上のバッファメモリを設けてもよい。

【 0 0 6 3 】 図 1 7 は、本発明のデータ圧縮・復元装置の実施例を示す図である。図において、データ圧縮・復元装置はデータ圧縮回路及びデータ復元回路から構成される。データ圧縮回路は中間データ変換手段 1、バッファメモリ B 1 及び符号化手段 2 から構成する。中間データ変換手段 1 は、入力データを中間データに変換する。バッファメモリ B 1 は、中間データ変換手段 1 から出力された中間データを一時的に格納する。符号化手段 2



は、中間データを出力符号に符号化する。

【 0 0 6 4 】また、データ復元回路は復号化手段 3、バッファメモリ B 2 及び中間データ逆変換手段 4 から構成する。復号化手段 3 は、入力符号を中間データに変換する。バッファメモリ B 1 は、復号化手段 3 から出力された中間データを一時的に格納する。中間データ逆変換手段 4 は、中間データを出力データに逆変換する。

【 0 0 6 5 】この装置により、データの圧縮処理及び復元処理の処理速度が向上する。なお、図 8 及び図 1 6 で示した実施例のように、データ圧縮回路では中間データ変換手段とバッファメモリ、データ復元回路ではバッファメモリと中間データ逆変換手段とを複数組設けることにより、より装置全体の処理速度を向上させることができる。また、バッファメモリは図 1 7 に図示した位置に限ることなく、図 2、図 4、図 1 0 及び図 1 2 で示した位置に、処理するデータの性質に応じて設けることにより、最適なデータ圧縮・復元装置を構成することができる。

【 0 0 6 6 】次に、上記の実施例で示したランレングス変換・逆変換、ランレングス変換手段及びランレングス逆変換手段について説明する。図 1 8 はランレングス変換・逆変換を説明する図である。ファクシミリ等において光学的に得られた画像データ 2 1 は、複数ライン（ここでは 4 ライン）のビットストリームで構成され、各ビットは白黒二値データで表される。この画像データ 2 1 は、各列を構成するビットの白黒のパターン及びそのパターンが二次元的に連続する数、すなわちランレングスに変換される。

【 0 0 6 7 】図において、例えば、左端の列 2 1 A を構成するビットは上から順に白白白黒である。この白白白黒のパターンは 2 列分連続するので、そのランレングスは 2 となる。この 2 列分のデータは変換データ 2 2 の左端の列 2 2 A で示すパターンとランレングスのデータとなる。なお、このパターンのデータ及びランレングスのデータをそれぞれ 4 ビットで表せば、ランレングスデータは 8 ビットで構成される。

【 0 0 6 8 】図 1 9 はランレングス変換手段の原理説明図である。図において、ランレングス変換手段は、データ入力手段 3 1、パラレルシリアル変換手段 3 2、ランレングス検出手段 3 3 及びランレングス出力手段 3 4 から構成される。

【 0 0 6 9 】データ入力手段 3 1 は、複数ラインのビットストリームで構成される入力データを情報源から入力する。パラレルシリアル変換手段 3 2 は、入力された入力データをパラレルからシリアルに変換する。すなわち、図 1 8 において示したように、列ごとに入力された入力データを行ごとのデータに変換する。ランレングス検出手段 3 3 は、そのシリアルに変換された入力データを 1 ビットずつシフトし、先行するビットとの間での入力データの一致を判別することによって、複数ラインの

同一列を構成するビットの入力データのパターン及びランレングスを検出する。ランレングス出力手段 3 4 はランレングス検出手段 3 3 が検出したランレングスデータを出力する。

【 0 0 7 0 】図 2 0 はランレングス逆変換手段の原理説明図である。図において、ランレングス逆変換手段は、ランレングスデータ入力手段 4 1、シリアル逆変換手段 4 2 及びデータ出力手段 4 3 から構成される。

【 0 0 7 1 】ランレングスデータ入力手段 4 1 は、情報源からランレングスデータを入力する。シリアル逆変換手段 4 2 はそのランレングスデータに基づいて、ランレングスデータをビットストリームで構成されるライン毎のシリアルデータに逆変換する。データ出力手段 4 3 はそのライン毎のシリアルデータを複数ラインの平行な出力データとして出力する。

【 0 0 7 2 】次に、上記の実施例で示した L Z W 符号化手段及び L Z W 復号化手段について、具体的な処理手順等を説明する。図 2 1 は、入力文字列を L Z W 符号の符号化アルゴリズムによって符号化する場合の具体例を示す図である。この入力文字列は、a、b、c の 3 文字だけの組み合わせからなる文字列である。まず、予め辞書に、1 文字の a、b、c だけをそれぞれ符号 1、2、3 に対応づけて登録する初期化を行う。

【 0 0 7 3 】まず、入力文字列 7 1 を左から右へ一字ずつ読み込む。最初の文字 a を読み込み、この a を語頭文字（列）（prefix string）とする。次に、2 番目の文字 b を読み込み、先の語頭文字 a にこの b を加えた a b を辞書の登録文字列と照合する。このとき、a b に一致する文字列が辞書にないので、先の語頭文字 a の対応符号 1 を符号化出力として出力する。この出力される符号を、出力符号欄 7 2 に示す。同時に、文字列 a b を符号 4 に対応させて辞書に登録する。この辞書に登録される内容を、登録内容欄 7 3 に示す。ここで、改めて 2 番目の入力文字 b を語頭文字とする。

【 0 0 7 4 】次に、入力文字列 7 1 の 3 番目の文字 a を読み込み、語頭文字 b にこの a を加えた b a を辞書の登録文字列と照合する。このとき、b a に一致する文字列が辞書にはないので、語頭文字 b の対応符号 2 を符号化出力として出力するとともに、文字列 b a を符号 5 に対応させて辞書に登録する。また、改めて 3 番目の入力文字 a を語頭文字とする。

【 0 0 7 5 】さらに、4 番目の文字 b を読み込み、語頭文字 a にこの b を加えた a b を辞書の登録文字列と照合する。このとき、辞書には a b に一致する文字列が登録されているので、このときは a b を語頭文字列とする。さらに、5 番目の入力文字 c を読み込み、語頭文字列 a b にこの c を加えた a b c を辞書の登録文字列と照合する。このとき、a b c に一致する文字列が辞書にないので、語頭文字列 a b の対応符号 4 を符号化出力として出力するとともに、文字列 a b c を符号 6 に対応させて辞

書に登録する。そして、改めて 5 番目の入力文字 c を語頭文字とする。

【0076】以下、同様のアルゴリズムにより、符号化と辞書登録を続ける。このアルゴリズムで入力文字列 a, b, a, b, c, ... に対して符号化が行われ、図 21 の出力符号欄 72 に示すような符号 1, 2, 4, 3, ... が符号化出力として出力される。そして、図 23 (A) に示すような登録文字列 91 と対応符号 92 との対応関係が辞書に登録される。

【0077】図 22 は、図 21 に示した符号化の処理手順を示すフローチャートである。図において、S に続く数字はステップ番号を示す。

【S101】予め初期化によって、入力される可能性のある全一文字に対しそれぞれ符号を対応させて辞書に登録する。また、辞書において次に登録すべきアドレス n を、例えば 256 に設定する。ここで、n は辞書に登録される文字列に対応して符号を 0, 1, 2, ... と付した場合、登録文字列の総数に相当する。さらに、入力文字列を読み込み、入力した最初の文字を語頭文字列 (prefix string)  $\omega$  とする。

【0078】【S102】次の入力文字 K を読み込む。

【S103】ステップ S102 において、入力文字データが存在したか否かを判別する。もし、入力文字データが存在すればステップ S105 へ進み、存在しなければステップ S104 へ進む。

【0079】【S104】語頭文字列  $\omega$  を辞書と照合し、対応する符号 code ( $\omega$ ) を読み出し、符号化出力として出力する。このとき、符号 code ( $\omega$ ) のビット数が  $\lceil \log_2 n \rceil$  の 2 進数符号に変換して出力する。ここで、記号  $\lceil x \rceil$  は、数値 x 以上の整数のうち、最小の整数を表す。以下、この意味で記号  $\lceil x \rceil$  を用いることにする。なお、このステップでは処理すべき文字列がないため、本ステップを実行後、本処理手順を終了する。

【0080】【S105】語頭文字列  $\omega$  に、ステップ S102 で読み込んだ文字 K を加えた文字列  $\omega K$  を辞書と照合し、文字列  $\omega K$  が辞書に登録されているか否かを判別する。もし、登録されていればステップ S106 に進み、登録されていなければステップ S107 に進む。

【0081】【S106】文字列  $\omega K$  を改めて語頭文字列  $\omega$  とする。そして、再びステップ S102 に戻る。このように、ステップ S102 ~ ステップ S106 を繰り返すことにより、入力文字列と一致する文字列として、辞書に登録された文字列のうちの最大長の文字列が探索される。

【0082】【S107】語頭文字列  $\omega$  を辞書と照合し、対応する符号 code ( $\omega$ ) を読み出し、符号化出力として出力する。このときの符号 code ( $\omega$ ) のビット数は、 $\lceil \log_2 n \rceil$  となる。また、文字列  $\omega K$  に n の値を対応させて辞書に登録する。すなわち、辞書のアドレス n に文字列  $\omega K$  を記憶する。さらに、ステップ S102 で

読み込んだ文字 K を語頭文字列  $\omega$  とするとともに、辞書アドレス n をインクリメントして、つぎの新たな入力文字列に対するステップ S102 以降の実行に備える。

【0083】図 24 は、図 21 に示した符号化出力を復元する場合の具体例を示す図である。予め復元側の辞書に、初期化によって、符号 1, 2, 3 だけがそれぞれ文字 a, b, c に対応づけられて登録されている。

【0084】まず、入力符号 81 を左から右へ一つずつ読み込む。最初の符号 1 を読み、辞書を参照して文字列 a を復元する。このとき復元された文字列を、復元文字列欄 821 に示す。最初の符号は、初期化によって必ず辞書に登録されている。そして、2 番目の符号 2 を読み、辞書を参照して文字列 b を復元する。このとき、前回の入力符号 1 と今回復号した文字列の最初の一文字 b とを組み合わせた「1b」に符号 4 を対応させて辞書に登録する。このときの登録された内容を、登録内容欄 83 に示す。

【0085】次に、入力文字列 81 の 3 番目の符号 4 を読み、辞書を参照して対応する「1b」を読み出す。さらに、「1b」の符号 1 を、辞書を参照して対応する文字 a を読み出す。このような一連の読み出し繰り返し動作を「再帰的な復号」と呼ぶ。これを、再帰的復号欄 82 に示す。これによって、文字列 a b が復元され、復元文字列として出力する。出力される文字列を、復元文字列欄 831 に示す。同時に、前回の入力符号 2 と今回復元した文字列の最初の一文字 a とを組み合わせた「2a」に、符号 5 を対応させて辞書に登録する。

【0086】以下、同様のアルゴリズムにより文字列の復元と辞書登録を続ける。このようにして入力符号 1, 2, 4, 3, 5, ... に対して復元が行われ、図 24 の復元文字列欄 821 に示すような文字列 a, b, a b, c, b a, ... が復元文字列として出力される。そして、図 23 (B) に示すような登録符号 93 と対応文字列 94 との対応関係が辞書に登録される。

【0087】図 25 は、図 24 に示した復号化の処理手順を示すフローチャートである。図において、S に続く数字はステップ番号を示す。

【S111】予め初期化によって、入力される可能性のある符号に対しそれぞれ文字を対応させて辞書に登録する。また、辞書において次に登録すべきアドレス n を、例えば 256 に設定する。ここで、n は辞書に登録される文字列に対応して符号を 0, 1, 2, ... と付した場合、登録文字列の総数に相当する。次に、入力符号を読み込み、最初の入力符号 CODE (バイナリコード) を 10 進数の入力符号  $\omega$  に変換する。この場合、図 22 の符号化では  $\omega$  は入力文字列であったが、この復号化では  $\omega$  は入力符号である点に注意されたい。そして、この  $\omega$  を OLD  $\omega$  とする。同時に、最初に入力する符号は既に辞書に登録されているため、入力符号  $\omega$  に対応する文字 D ( $\omega$ ) を辞書から探索し、復元された文字として出力す

る。なお、出力した文字を後述するステップS 1 1 6 の例外処理のためにFINchar にセットしておく。

【0088】〔S 1 1 2〕次の入力符号CODEを読み込む。

〔S 1 1 3〕ステップS 1 1 2において入力符号データが存在したか否かを判別する。もし、存在すればステップS 1 1 5へ進み、存在しなければ本処理手順を終了する。

【0089】〔S 1 1 4〕読み込んだ入力符号CODEから入力符号 $\omega$ に変換するとともに、この入力符号 $\omega$ をIN $\omega$ にセットする。

〔S 1 1 5〕入力符号 $\omega$ をnと比較する。すなわち、入力符号が辞書に登録されているか否か ( $\omega \geq n$ ) を判別する。もし、 $\omega$ がnより小さいときにはステップS 1 1 7へ進み、 $\omega$ がn以上のときにはステップS 1 1 6へ進む。なお、 $\omega$ がn以上になるのは、例えば図24の入力符号欄81が「8」のときである。

【0090】〔S 1 1 6〕ステップS 1 1 1または前回にステップS 1 1 9で設定されたOLD  $\omega$ およびFINcharの組 (OLD  $\omega$ , FINchar) を $\omega$ Kと置き換える。すなわち、OLD  $\omega$ にセットされた値を $\omega$ に、FINcharにセットされた値をKにセットする。そして、Kをスタックにプッシュ (PUSH) する。なお、 $\omega$ はステップS 1 1 7で復号化される。

【0091】〔S 1 1 7〕通常、入力符号 $\omega$ は前回までの処理で辞書に登録されているため、入力符号 $\omega$ に対応する文字列D ( $\omega$ ) を辞書から読み出す。読み出した文字列D ( $\omega$ ) を $\omega_i$  Kに分解する。 $\omega_i$ は符号、Kは復号化文字である。そして、文字列D ( $\omega$ ) が、 $\omega_i$  Kに分解できない1文字であるか否かを判別する。D ( $\omega$ ) が $\omega_i$  Kに分解できるならばステップS 1 1 8に進み、D ( $\omega$ ) が1文字であるならばステップS 1 1 9へ進む。

【0092】〔S 1 1 8〕文字Kを一時的にスタックにプッシュし、また符号 $\omega_i$ を新たな $\omega$ とし、再度ステップS 1 1 7に戻る。このステップS 1 1 7およびステップS 1 1 8の実行を、D ( $\omega$ ) が1文字に至るまで繰り返す。

【0093】〔S 1 1 9〕ステップS 1 1 8でスタックにプッシュした各文字をLIFO (Last In First Out) 形式でポップ (POP) して復元文字列を出力する。例えば、図24の入力符号欄81が「5」の場合ならば、a, bの順でスタックにプッシュされ、baという復元文字列が出力される。同時に、今回復元した文字列の最初の一文字をFINchar とし、前回セットされた OLD $\omega$ とFINchar との組 (OLD $\omega$ , FINchar) からなる文字列を、nの値に対応させて辞書に登録する。すなわち、この文字列を辞書のアドレスnに記憶する。さらに、nをインクリメントし、ステップS 1 1 4でセットされたIN $\omega$ をOLD  $\omega$ にセットして、次のステップS 1 1 2以降の

実行に備える。

【0094】上述のように復号化処理では、図25のステップS 1 1 7～ステップS 1 1 9を繰り返し行うことによって符号化前のデータに復元する。すなわち、入力符号 $\omega$ は前回までの処理で辞書に登録されているため、入力符号 $\omega$ に対応する文字列D ( $\omega$ ) を辞書から読み出す。また、読み出した文字列D ( $\omega$ ) を $\omega_i$  Kに分解し、この文字Kを一時的にスタックに退避させる。そして、符号 $\omega_i$ を新たな入力符号 $\omega$ として、再度入力符号 $\omega$ に対応する文字列D ( $\omega$ ) を辞書から読み出す。これらの手順を、新たな入力符号 $\omega$ が一文字になるまで再帰的に繰り返す。そして、スタックに退避させた文字をLIFO形式でポップして出力するという方式である。

【0095】図26は、上記処理手順等に使用した辞書の木構造の一例を示す図である。この辞書の木構造は、LZW符号化手段及びLZW復号化手段において実現されるアルゴリズムによる符号化及び復号化の際に用いられる辞書の内部構造を図示したものである。図26において、丸数字は識別番号を示し、この丸数字が付されている箇所を「ノード (node; 節)」と呼ぶ。

【0096】辞書50は、ルート (root; 根) 51を起点とする。このルート51には、文字は割り当てられない。そして、ルート51の一階層下、すなわち第1階層52には一文字目の文字が登録される。この一文字目の文字の登録は、相異なる文字が登録され、主に辞書50の初期化の時に行われる。図には「a」、「b」及び「c」の3文字が登録されているが、実際には8ビットのデータで表現可能な256文字が登録される。

【0097】そして、第2階層53から下の階層は、情報源から入力された文字列を学習することによって登録される文字である。なお、一つ下の階層を有するノードを「枝 (branch)」と呼び、一つ下の階層を有するノードを「葉 (leaf)」と呼ぶ。したがって、図では丸数字の25, 26, 13, 14, 27, 28, 16, 6, ... , 22, 23, 24のノードが「葉」であり、その他のノードは「枝」である。

【0098】なお、あるノードが現在は「葉」であっても、学習により「枝」となる可能性がある。例えば、

「acd」という文字列を辞書50に登録する場合、文字列「ac」は第1階層52が「a」(丸数字1)、第2階層53が「c」(丸数字6)として登録されているので、第2階層53の「c」の下の第3階層54に、新たに「d」を登録することになる。このとき、丸数字6のノードは「葉」から「枝」に変わる。

【0099】上記の実施例の説明では、データ圧縮回路において、符号は $\log_2 n$ 以上の最小の整数のビット数からなる出力符号で出力したが、本出願人が特願平3-130623号において開示したように、ビット端数補償、Phasing in Binary Codes、あるいは多値算術符号からなる出力符号で出力してもよい。

【 0 1 0 0 】また、符号化及び復号化の際に用いられる辞書は木構造に基づき構築したが、ハッシュ関数に基づき辞書を構築する等のように、他の構築法に基づき辞書を構築してもよい。例えば、二進分木（バイナリ・ツリー）法によって辞書を構築し、辞書に登録された文字列等のデータを二進探索（バイナリ・サーチ）により探索するようにしてもよい。

【 0 1 0 1 】なお、上記の各実施例は、ワークステーション等における文字コード、ベクトル情報、画像データなどの圧縮及び復元に応用され、必要な記憶容量を大幅に削減することができる。

【 0 1 0 2 】また、通信回線を利用したデータ送受信においても応用でき、通信時間の短縮を図ることができる。例えば、モデム、ファクシミリ等の通信機器に応用することができる。

【 0 1 0 3 】

【発明の効果】以上説明したように本発明では、データ圧縮回路において、他の装置と中間データ変換手段との間バッファメモリを設けたので、いずれもデータの性質による互いの処理速度の変動に影響されことなく変換処理することができる。

【 0 1 0 4 】また、データ圧縮回路において、中間データ変換手段と符号化手段との間バッファメモリを設けたので、いずれの手段もデータの性質によるたの手段の処理速度の変動に影響されことなく変換処理することができる。

【 0 1 0 5 】さらに、データ圧縮回路において、符号化手段と他の装置との間バッファメモリを設けたので、いずれもデータの性質による互いの処理速度の変動に影響されことなく変換処理することができる。

【 0 1 0 6 】そして、データ復元回路において、他の装置と復号化手段との間にバッファメモリを設けたので、いずれもデータの性質による互いの処理速度の変動に影響されことなく逆変換処理することができる。

【 0 1 0 7 】それから、データ復元回路において、復号化手段と中間データ逆変換手段との間にバッファメモリを設けたので、いずれの手段もデータの性質による他の手段の処理速度の変動に影響されことなく逆変換処理することができる。

【 0 1 0 8 】そのうえ、データ復元回路において、中間データ逆変換手段と他の装置との間にバッファメモリを設けたので、いずれもデータの性質による互いの処理速度の変動に影響されことなく逆変換処理することができる。

【 0 1 0 9 】そして、データ圧縮・復元装置において、データ圧縮回路では中間データ変換手段と符号化手段との間に、また、データ復元回路では復号化手段と中間データ逆変換手段との間に、バッファメモリを設けたので、いずれの手段もデータの性質による他の手段の処理速度の変動に影響されことなく変換処理及び逆変換処

理することができる。

【 0 1 1 0 】したがって、各回路及び装置全体の処理速度を向上させることができる。

【図面の簡単な説明】

【図 1】本発明のデータ圧縮回路の第 1 の原理説明図である。

【図 2】本発明のデータ圧縮回路の第 1 の実施例を示す図である。

【図 3】本発明のデータ圧縮回路の第 2 の実施例を示す図である。

【図 4】本発明のデータ圧縮回路の第 3 の実施例を示す図である。

【図 5】本発明のデータ圧縮回路の第 2 の原理説明図である。

【図 6】本発明のデータ圧縮回路の第 4 の実施例を示す図である。

【図 7】本発明のデータ圧縮回路の第 3 の原理説明図である。

【図 8】本発明のデータ圧縮回路の第 5 の実施例を示す図である。

【図 9】本発明のデータ復元回路の第 1 の原理説明図である。

【図 1 0】本発明のデータ復元回路の第 1 の実施例を示す図である。

【図 1 1】本発明のデータ復元回路の第 2 の実施例を示す図である。

【図 1 2】本発明のデータ復元回路の第 3 の実施例を示す図である。

【図 1 3】本発明のデータ復元回路の第 2 の原理説明図である。

【図 1 4】本発明のデータ復元回路の第 4 の実施例を示す図である。

【図 1 5】本発明のデータ復元回路の第 3 の原理説明図である。

【図 1 6】本発明のデータ復元回路の第 5 の実施例を示す図である。

【図 1 7】本発明のデータ圧縮・復元装置の原理説明図である。

【図 1 8】ランレングス変換・逆変換を説明する図である。

【図 1 9】ランレングス変換手段の原理説明図である。

【図 2 0】ランレングス逆変換手段の原理説明図である。

【図 2 1】符号化の具体例を示す図である。

【図 2 2】符号化の処理手順を示す図である。

【図 2 3】文字列と符号との対応関係図である。

【図 2 4】復号化の具体例を示す図である。

【図 2 5】復号化の処理手順を示す図である。

【図 2 6】辞書の木構造の一例を示す図である。

【図 2 7】従来の処理回路の概略ブロック図であり、

21

22

(A) は従来のデータ圧縮回路の概略ブロック図を、  
 (B) は従来のデータ復元回路の概略ブロック図をそれ  
 ぞれ示す。

【符号の説明】

1 中間データ変換手段

2 符号化手段

3 復号化手段

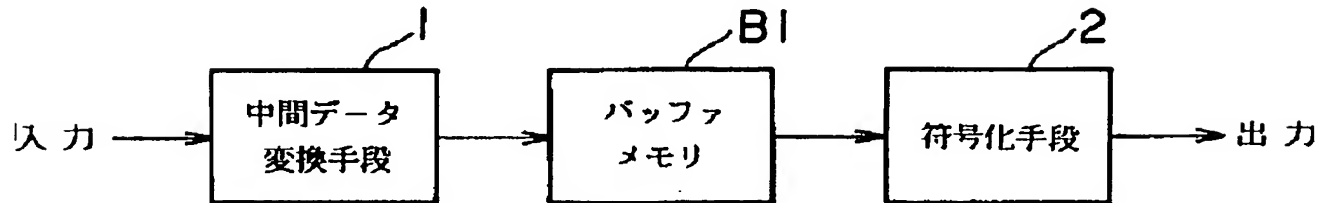
4 中間データ逆変換手段

B 1 バッファメモリ

B 2 バッファメモリ

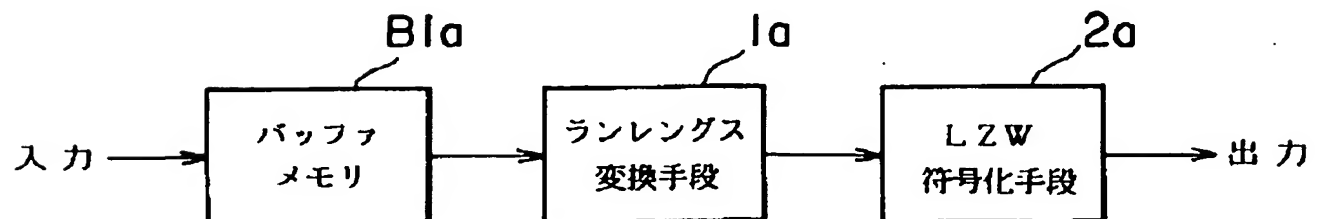
【図 1】

# 本発明のデータ圧縮回路の第 1 の原理説明図



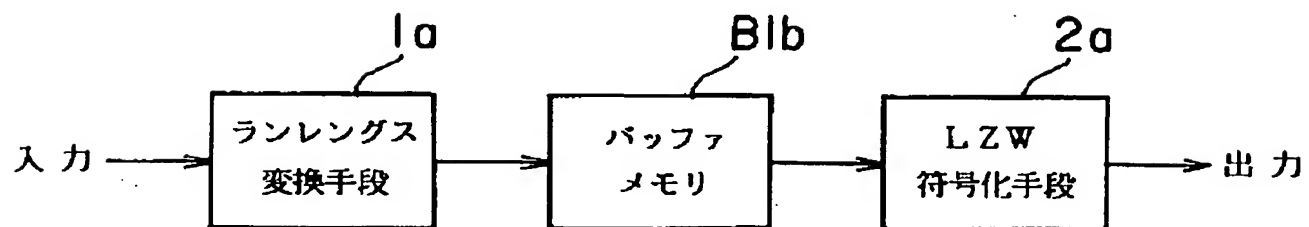
【図 2】

# 本発明のデータ圧縮回路の第 1 の実施例



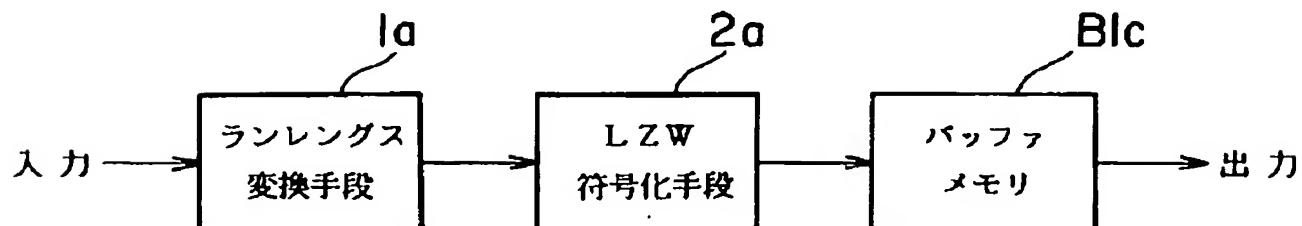
【図 3】

## 本発明のデータ圧縮回路の第 2 の実施例



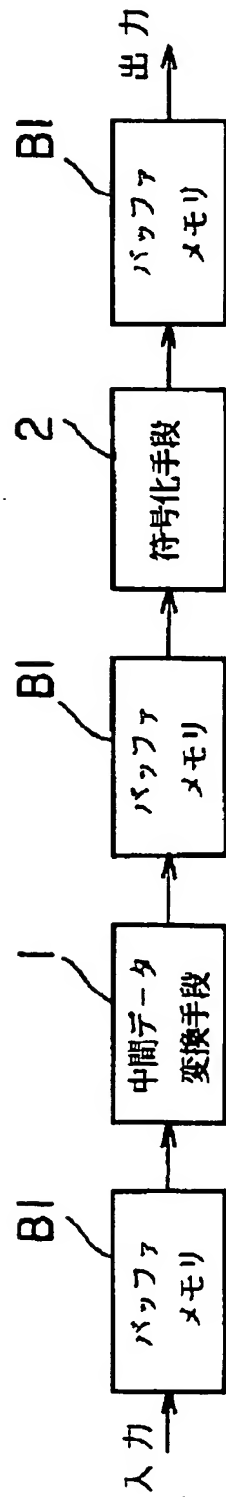
【図 4】

## 本発明のデータ圧縮回路の第 3 の実施例



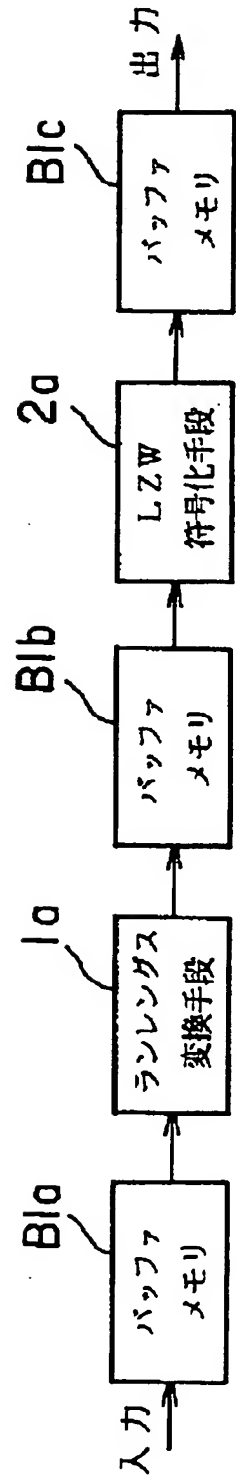
本発明のデータ圧縮回路の第2の原理説明図

【図5】

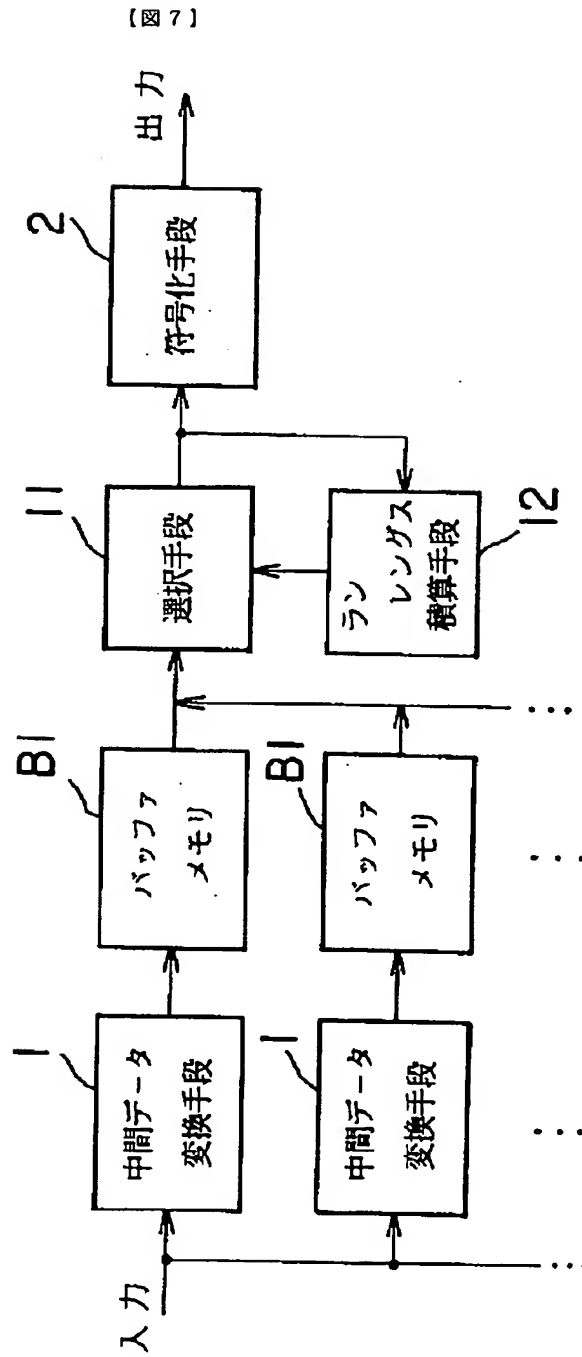


本発明のデータ圧縮回路の第4の実施例

【図6】

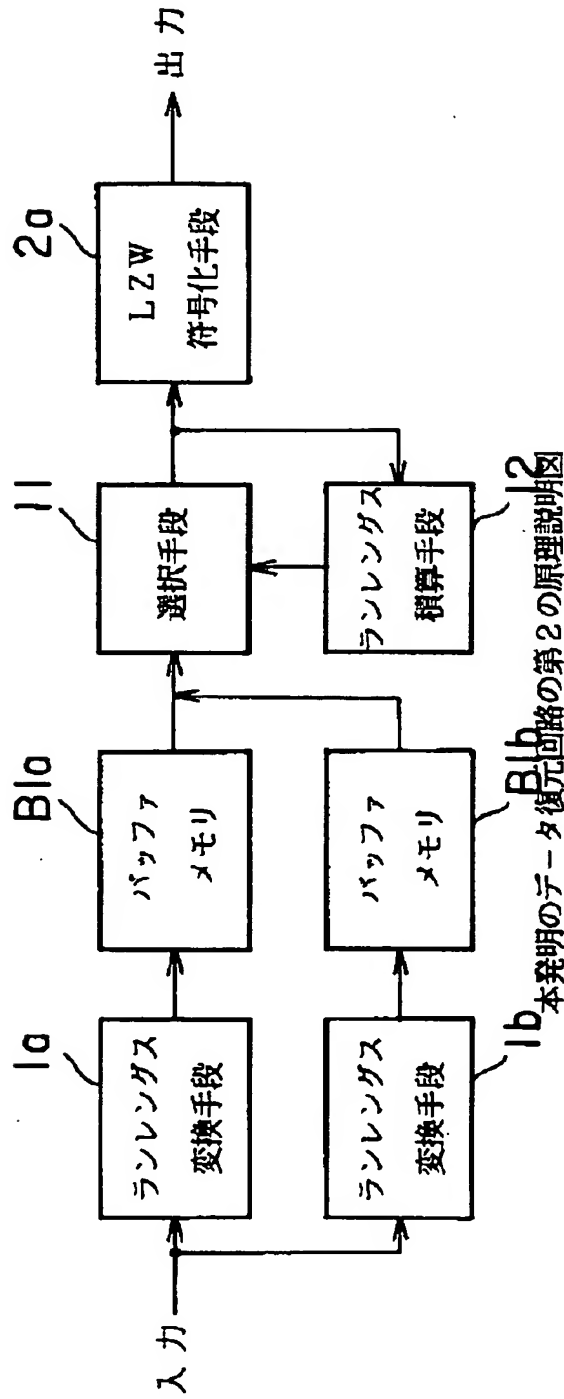


本発明のデータ圧縮回路の第3の原理説明図

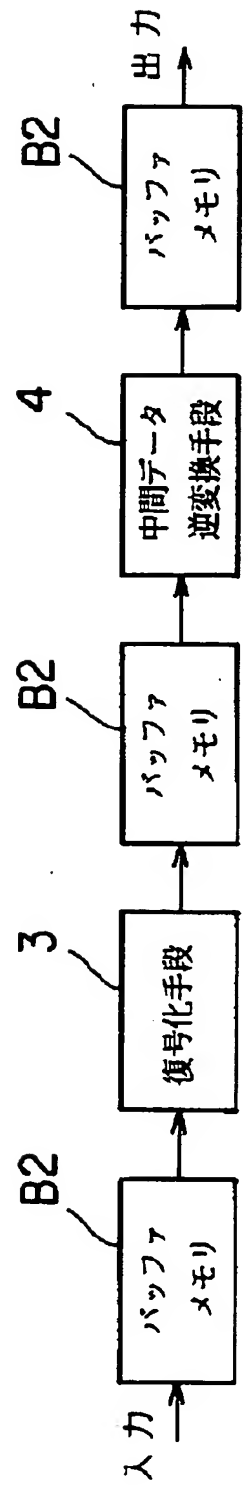




本発明のデータ圧縮回路の第5の実施例

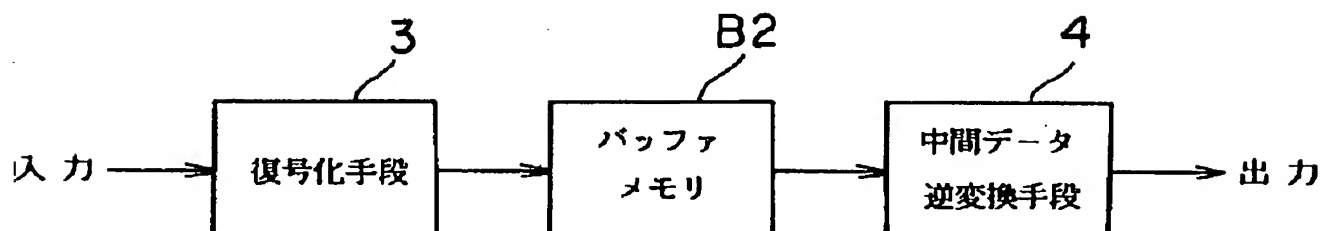


【図 13】



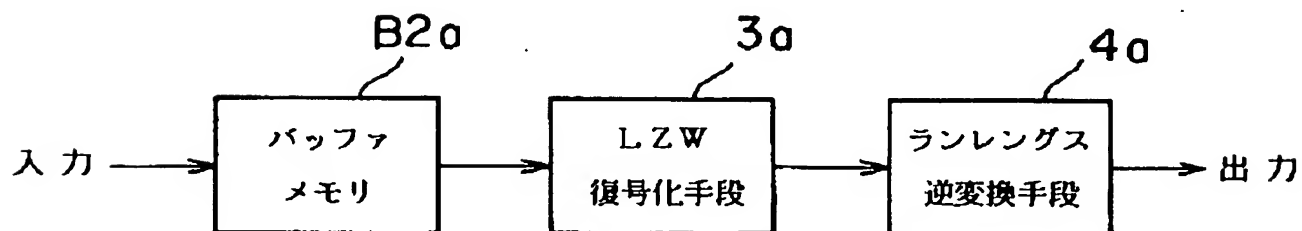
【図 9】

## 本発明のデータ復元回路の第 1 の原理説明図



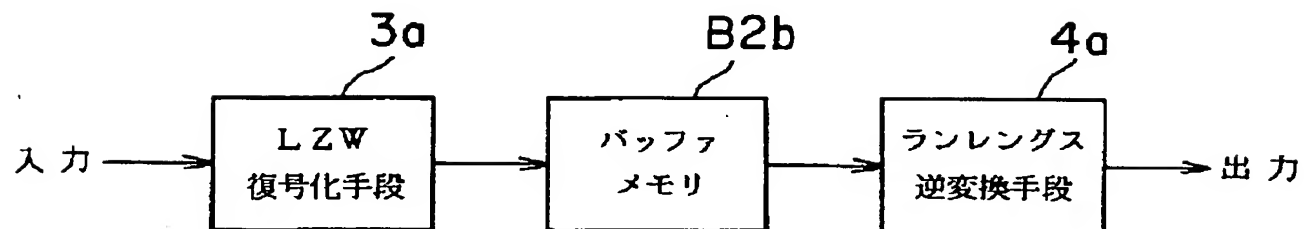
【図 10】

## 本発明のデータ復元回路の第 1 の実施例



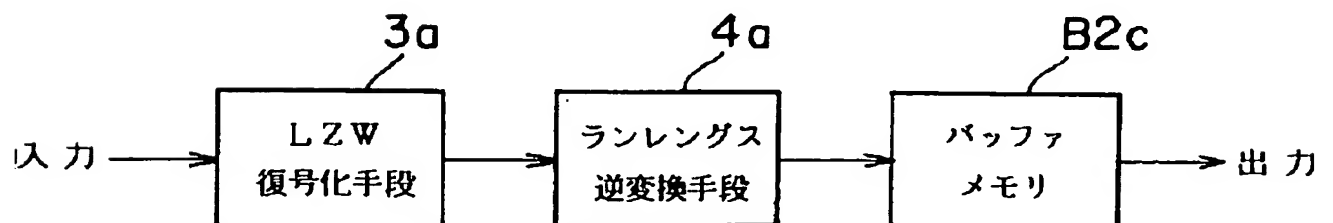
【図 1 1】

## 本発明のデータ復元回路の第 2 の実施例



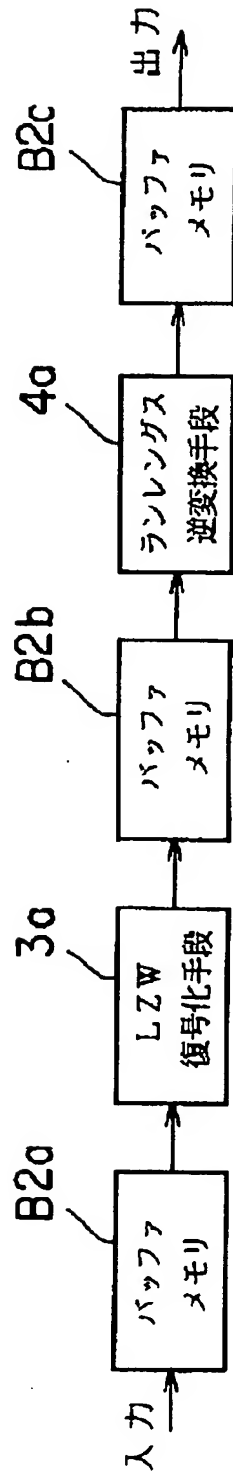
【図 1 2】

## 本発明のデータ復元回路の第 3 の実施例



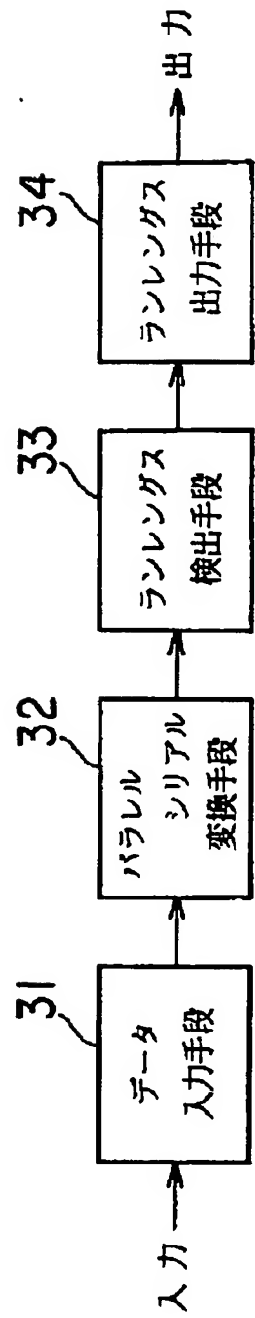
本発明のデータ復元回路の第4の実施例

【図14】



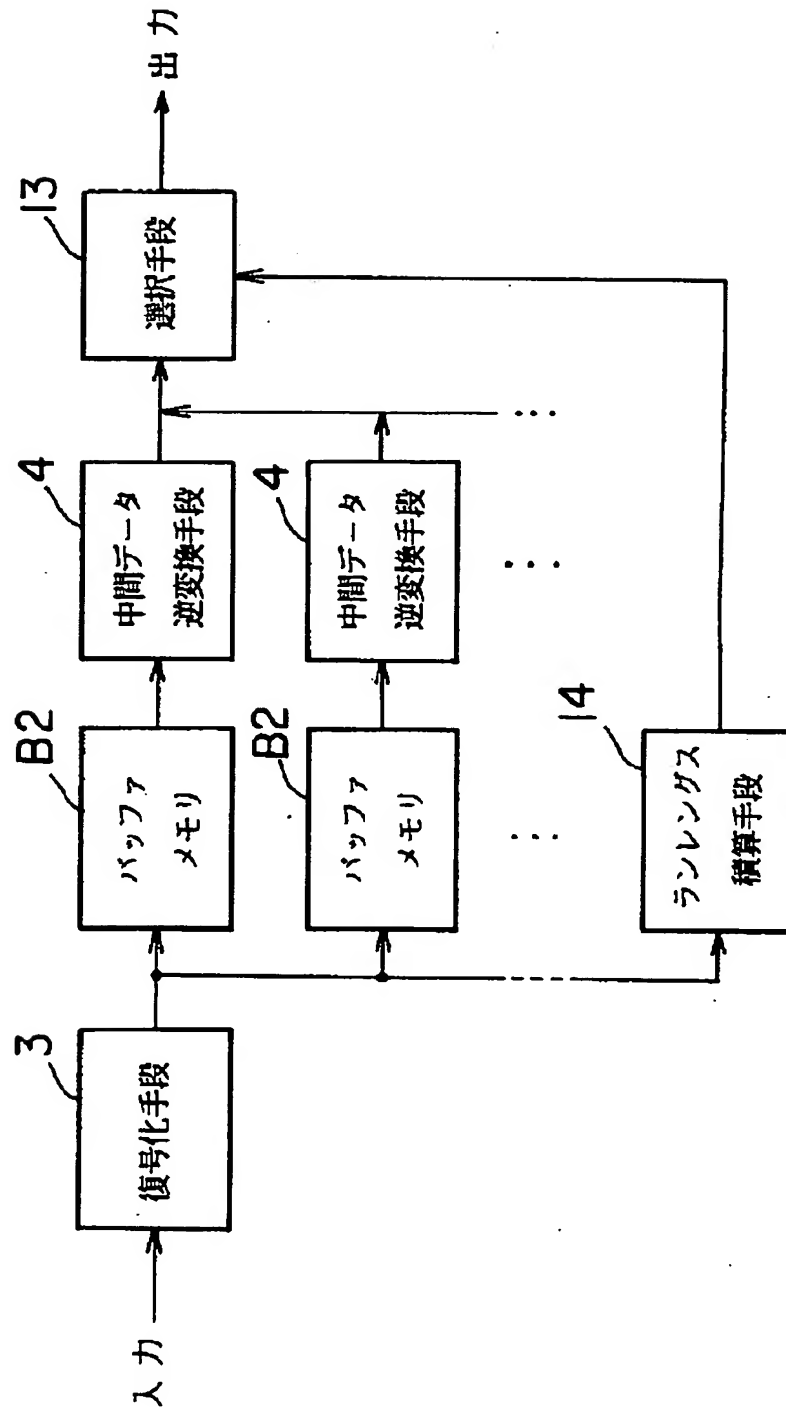
ランレングス変換手段の原理説明図

【図19】



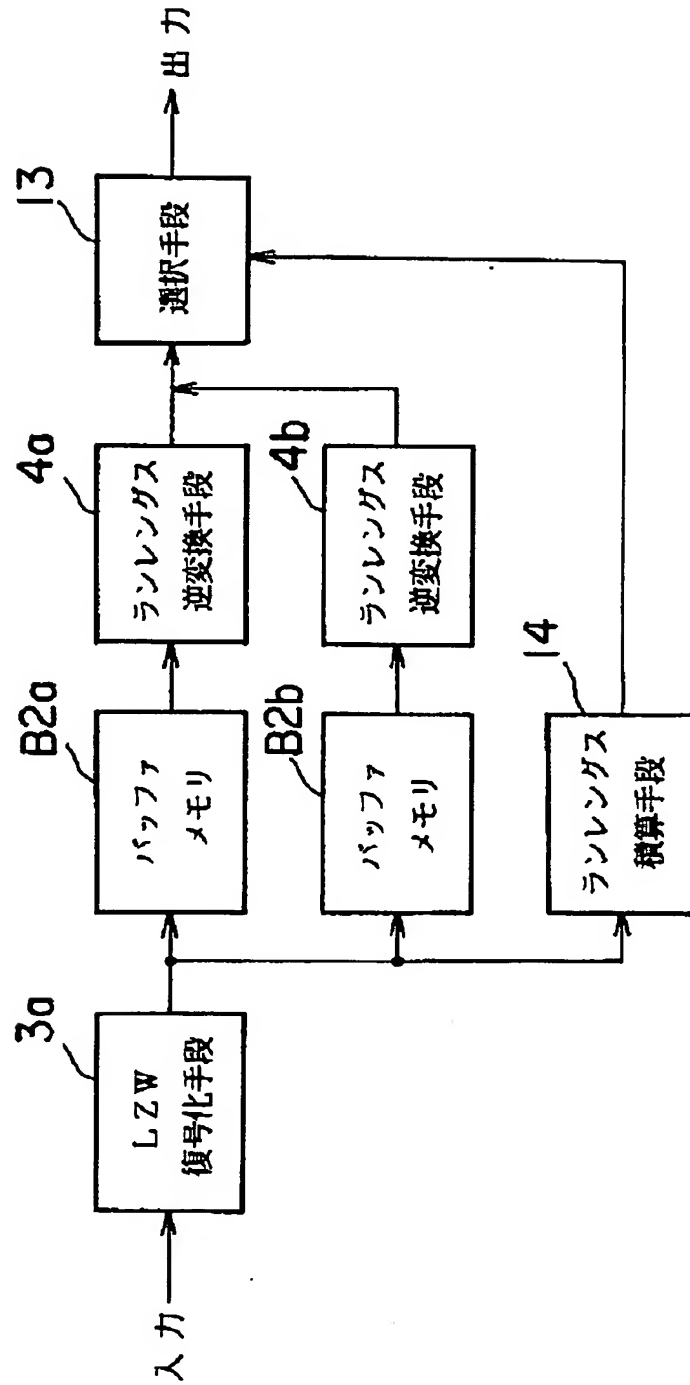
【図 1 5】

本発明のデータ復元回路の第3の原理説明図



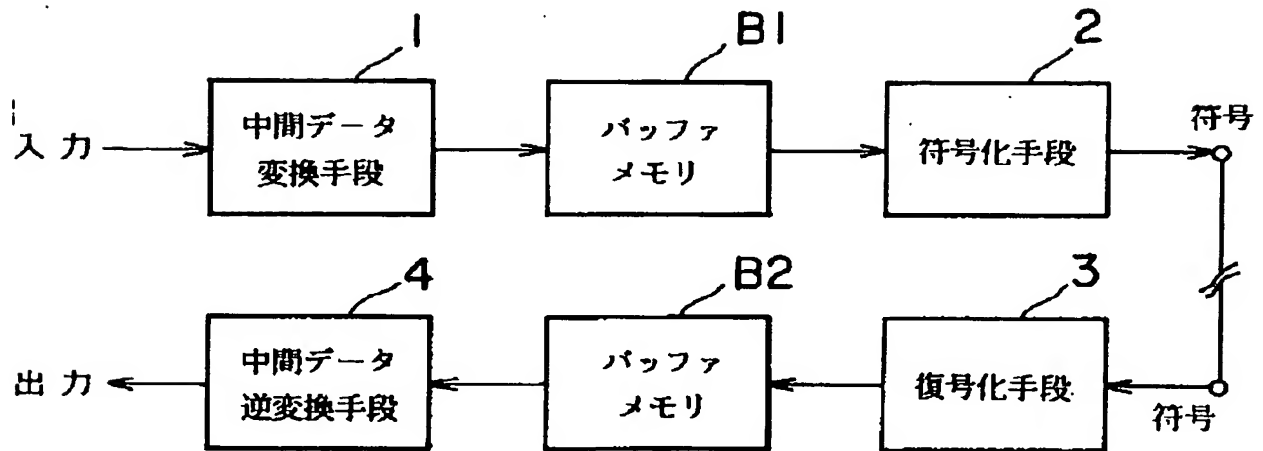
【図 16】

本発明のデータ復元回路の第5の実施例



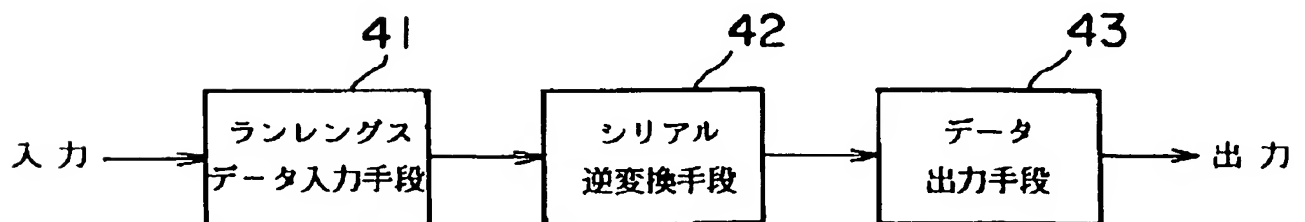
【図 17】

## 本発明のデータ圧縮・復元装置の原理説明図



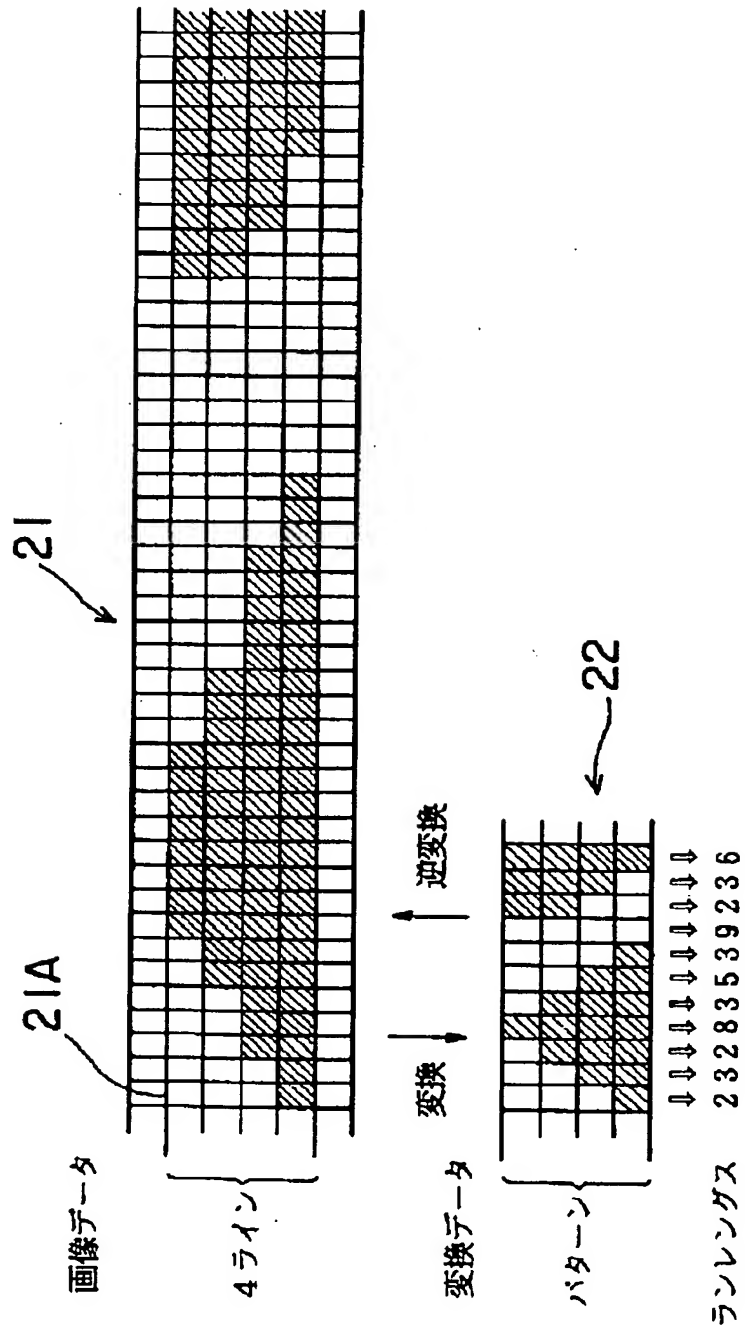
【図 20】

## ランレングス逆変換手段の原理説明図



【図 1 8】

ランレングス変換・逆変換を説明する図

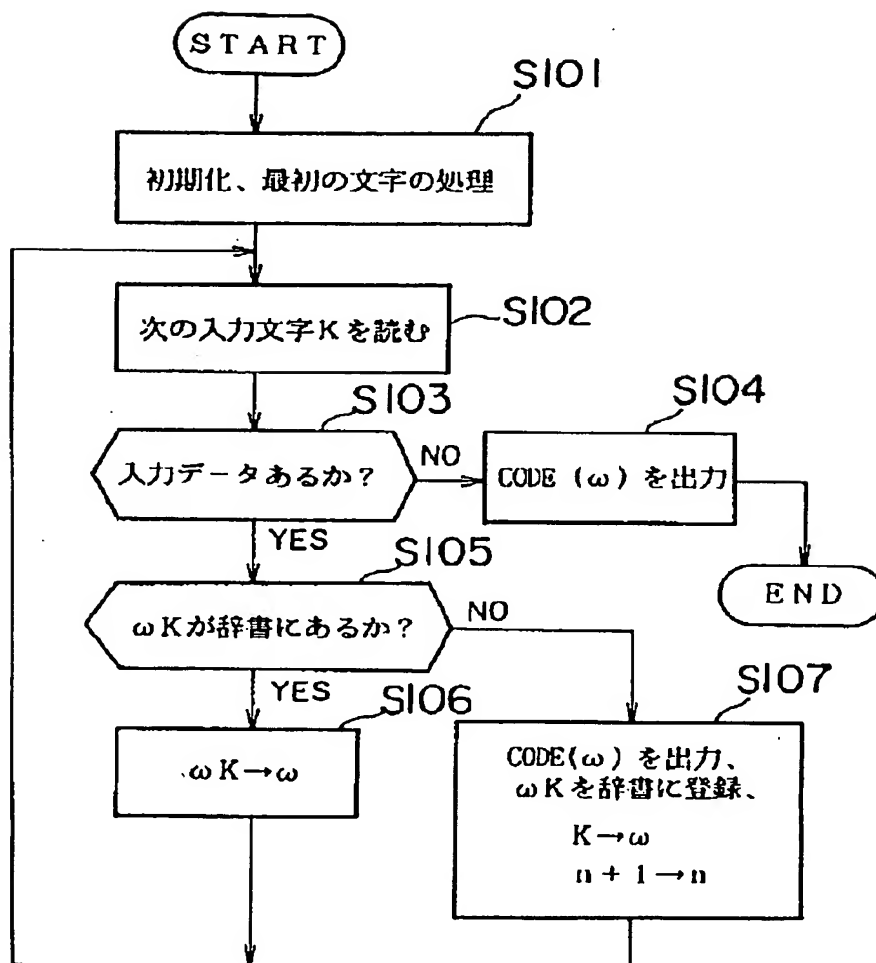






【図 2 2】

符号化の処理手順を示す図



【図 2 3】

## 文字列と符号との対応関係図

(A)		(B)	
登録文字列	対応符号	登録符号	対応文字列
a	1	1	a
b	2	2	b
c	3	3	c
ab	4	4	1b
ba	5	5	2a
abc	6	6	4c
cb	7	7	3b
bab	8	8	5b
baba	9	9	8a
aa	10	10	1a
aaa	11	11	10a
aaaa	12	12	11a

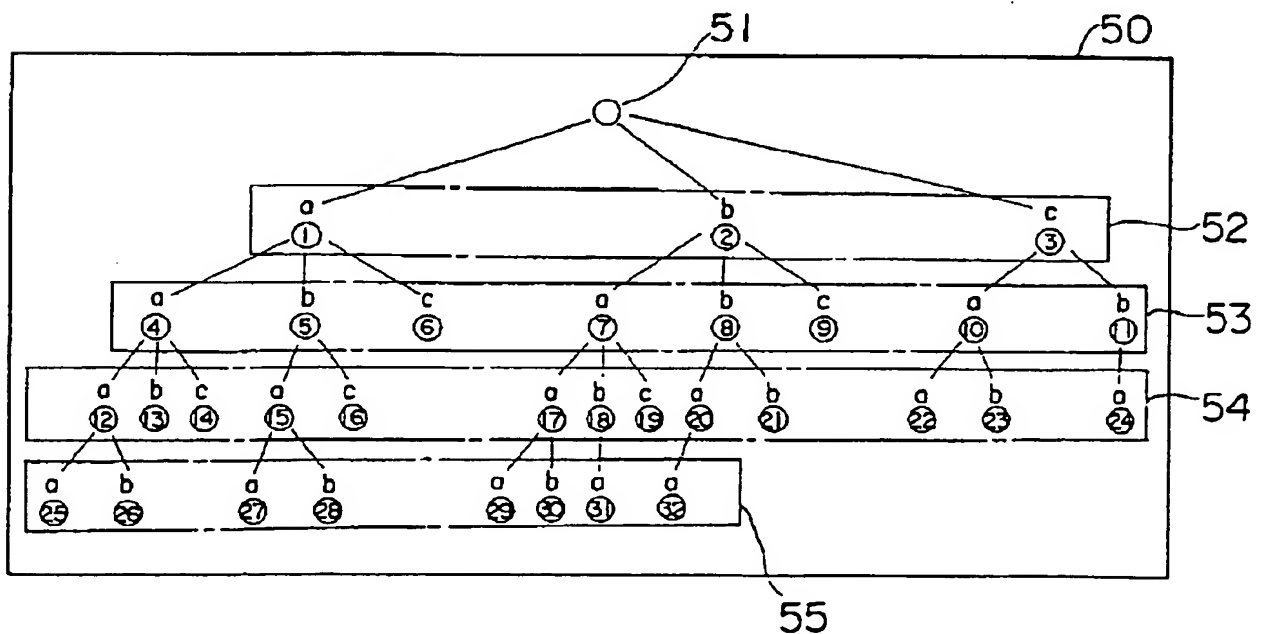
【図 2 4】

復号化の具体例を示す図

81	入力符号	1	2	4	3	5	8	1	10	11
	再帰的 復号化 ↓	a	b	1b	c	2a	5b	a	1a	10a
				a		b	2a		a	1a
							b			a
82	復元文字列	a	b	ab	c	ba	bab	a	aa	aaa
83	登録内容 (初期化 1→a 2→b 3→c)	4                  6                  8                  10								
		5                  7                  9                  11								

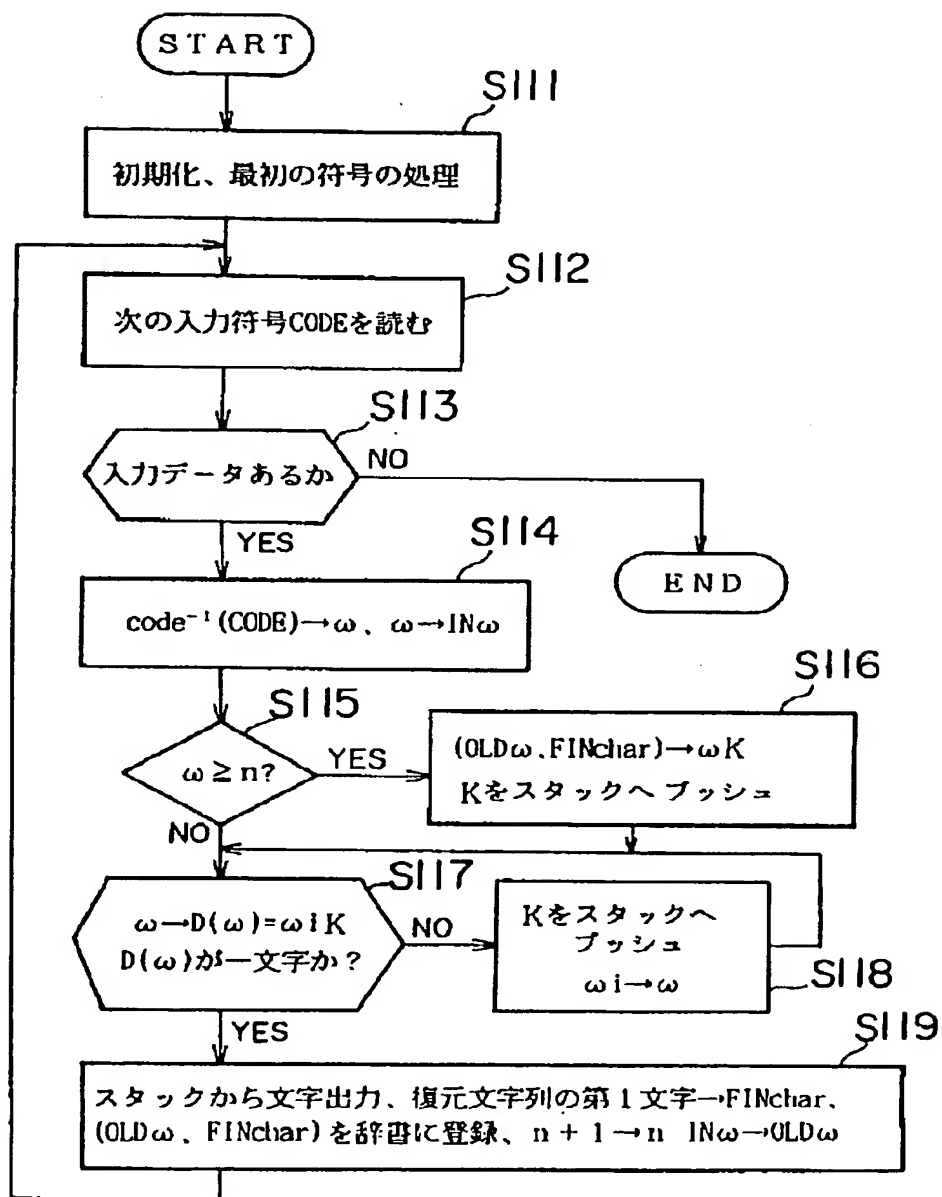
【図 2 6】

辞書の木構造の一例を示す図



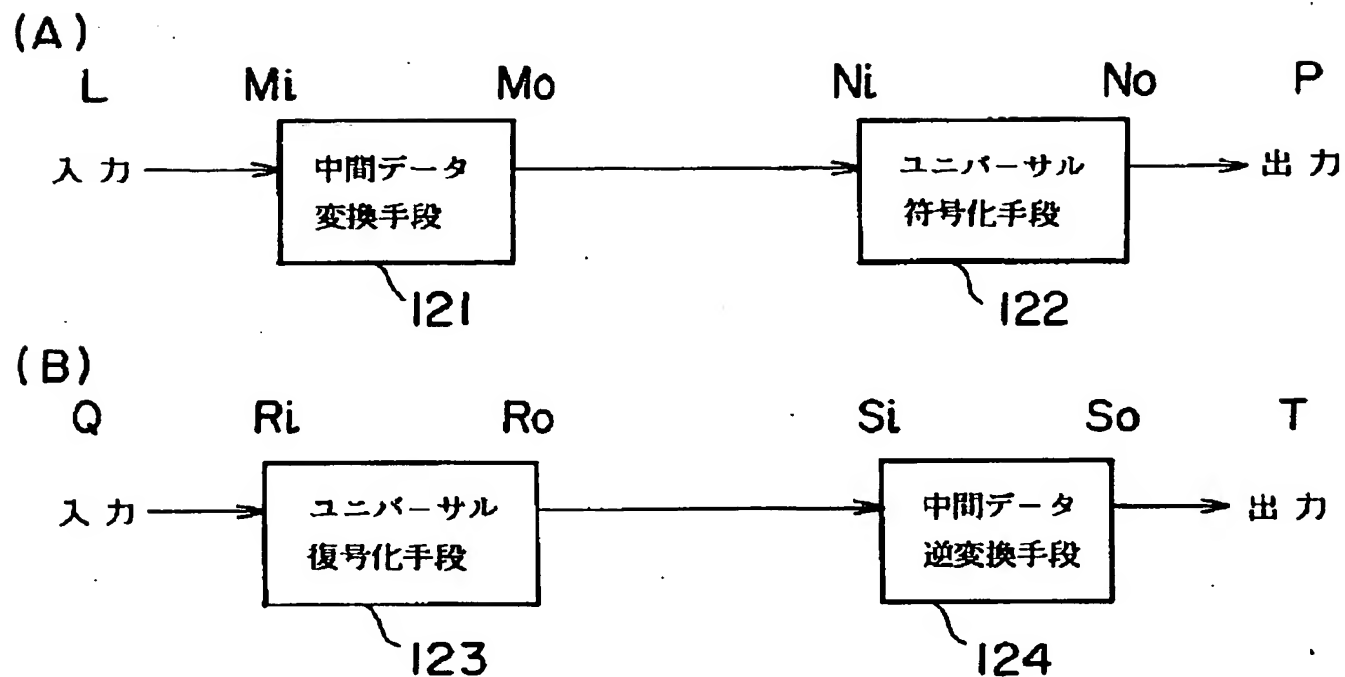
【図 2 5】

復号化の処理手順を示す図



【図 27】

## 従来の処理回路の概略ブロック図



---

フロントページの続き

(72) 発明者 中野 泰彦  
神奈川県川崎市中原区上小田中 1015 番  
地 富士通株式会社内

(72) 発明者 千葉 広隆  
神奈川県川崎市中原区上小田中 1015 番  
地 富士通株式会社内